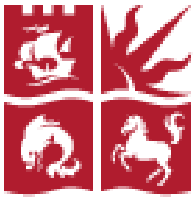
Trevor Spiteri, George Vafiadis, Jose Luis Nunez-Yanez
2009-09-02

A Toolset for the Analysis and Optimization of Motion Estimation Algorithms and Processors

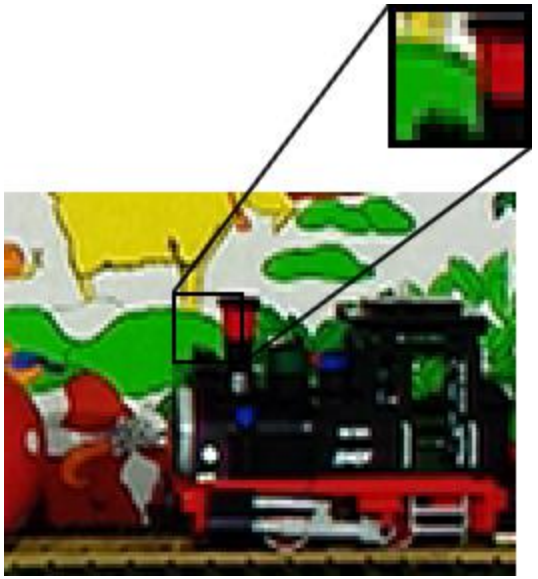


University of BRISTOL

Department of Electrical and Electronic Engineering

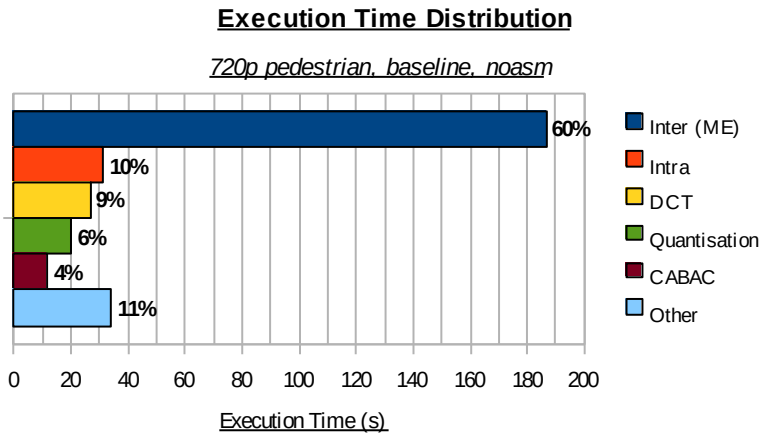
Overview

- Motion estimation takes time, full search expensive for HD.
- Flexible reconfigurable processor.
- IDE to design and test algorithms.
- Toolset to configure the processor.



Saving time

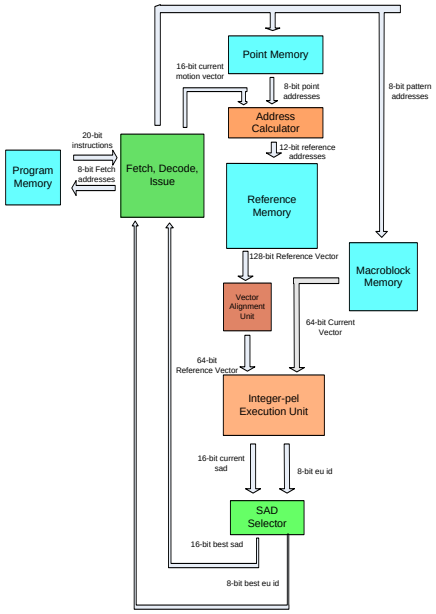
- Motion estimation takes processor time.
- The design space to explore is large.
- Configuring the ME processor takes developer times.



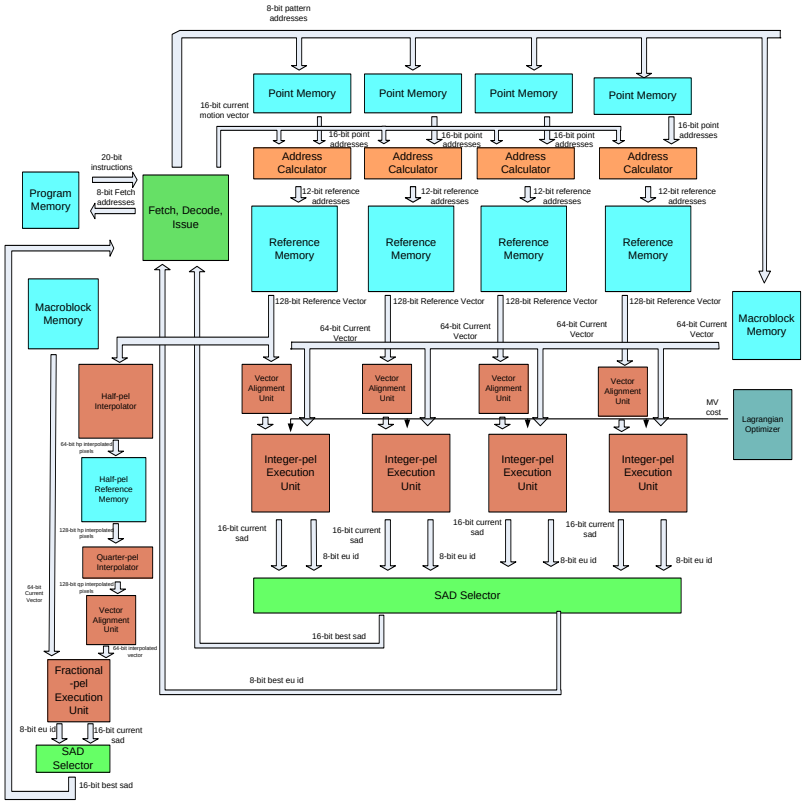
The reconfigurable processor

- Advanced features such as rate distortion optimization using Lagrangian techniques.
- Multiple motion vector candidates allowed.
- Multiple sub-partition sizes allowed.
- Multiple reference frames allowed.
- Can do fractional pel motion estimation that can be used for the H.264 standard.

Simple and complex configurations



Simple (1 integer pel unit)



Complex (4 int. pel units, 1 frac. pel unit, Lagrangian optimizer)

Processor performance and complexity evaluation

Processor Configuration	Speed (cycles/MB, frames/second)	FPGA size (LUTs, slices)	Memory (BRAMS)
Base configuration (1 integer-pel execution unit)	625 cycles/MB, 39 fps	2289 LUTs, 1300 slices	21 BRAMS (2 ref. areas, 112×128 pixels)
Complex configuration (4 integer-pel execution units)	234 cycles/MB, 104 fps	7074 LUTs, 3703 slices	72 BRAMS (2 ref. areas, 112×128 pixels)

Video sequence: 1080p *crowdrun* from SVT HD multi format test set

FPGA part: Virtex-4 SX35, 200 MHz clock frequency

Algorithm: 6-point hexagonal search (up to 8 steps), then 8-point square

Designing block-matching algorithms

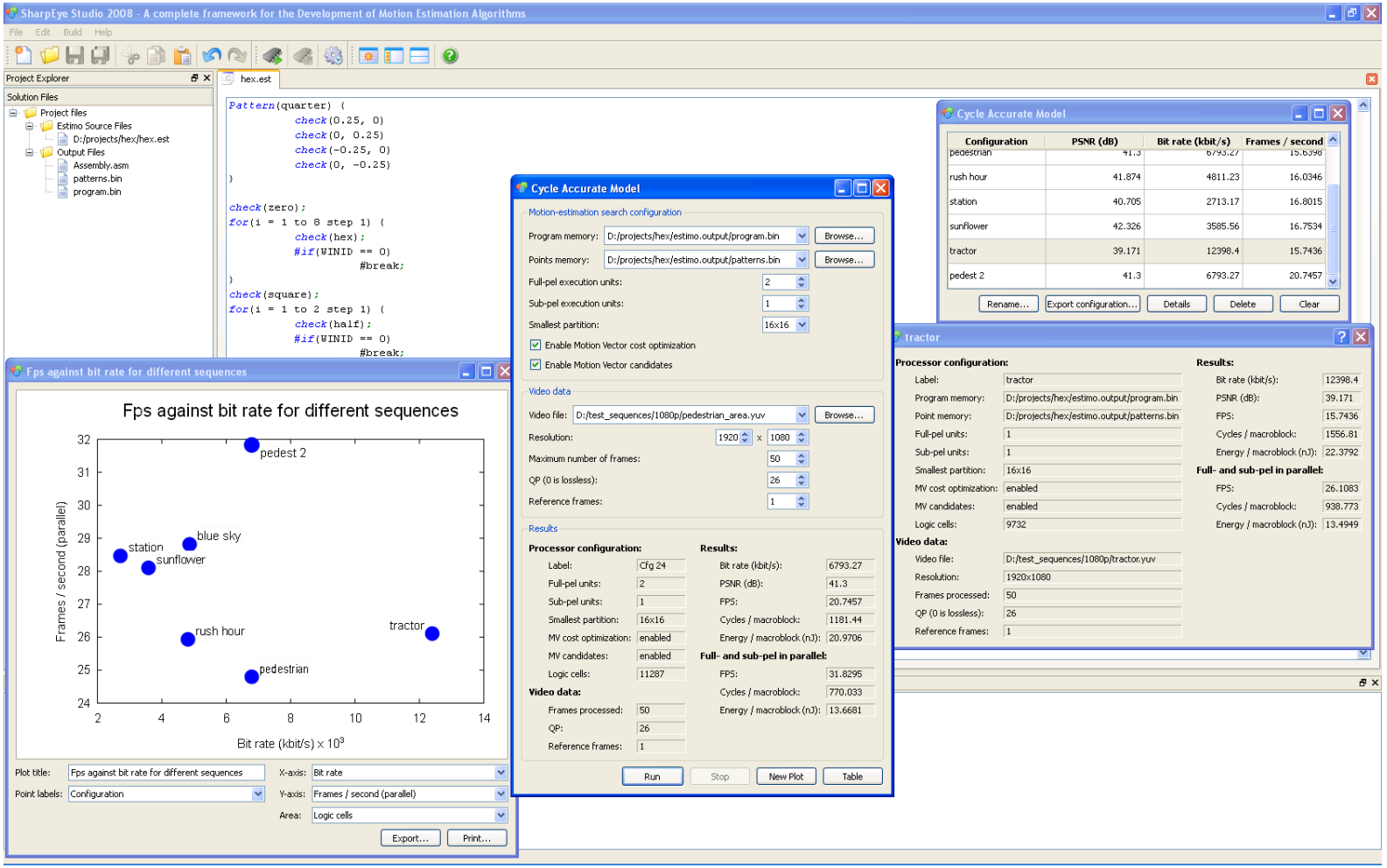
- Estimo C: high-level language for search algorithms.
- Compiler targets the reconfigurable processor.
- No need to know how hardware works.
- Compiled program works across all configurations.

```
s = 8; // initial step size
check(0, 0);
check(0, s);
check(0, -s);
check(s, 0);
check(-s, 0);
update;
do {
    s = s / 2;
    for (i = 1 to 5 step 1) {
        check(0, s);
        check(0, -s);
        check(s, 0);
        check(-s, 0);
        update;
        #if (WINID == 0)
            #break;
    }
} while (s > 1);
for (x = -0.5 to 0.5 step 0.25)
    for (y = -0.5 to 0.5 step 0.25)
        check(x, y);
update;
```

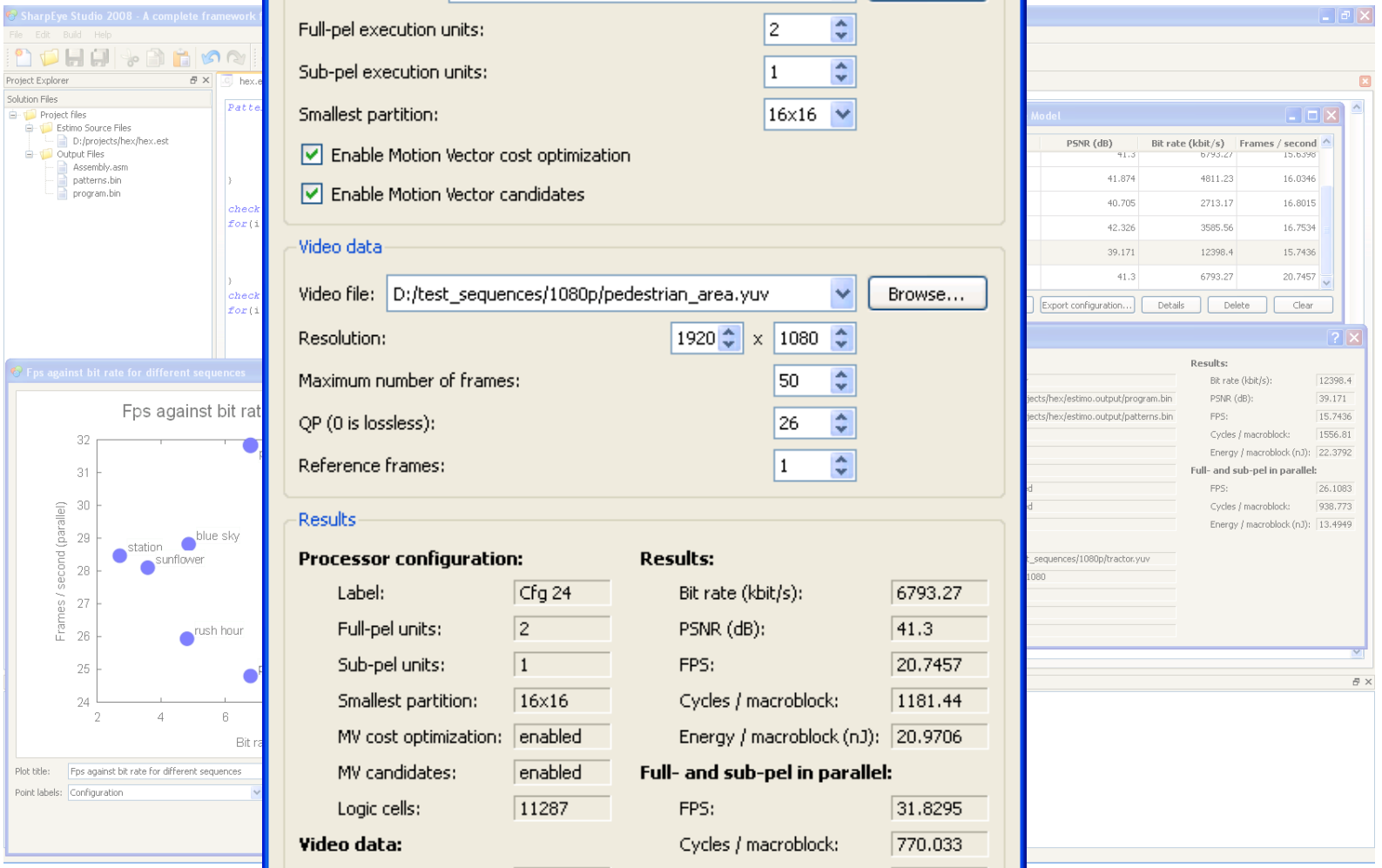
Cycle-accurate simulator

- Analysing processor configurations on hardware takes time.
- Using simulator, no need for synthesizing hardware and configuring board.
- No hardware required for evaluation of processor.

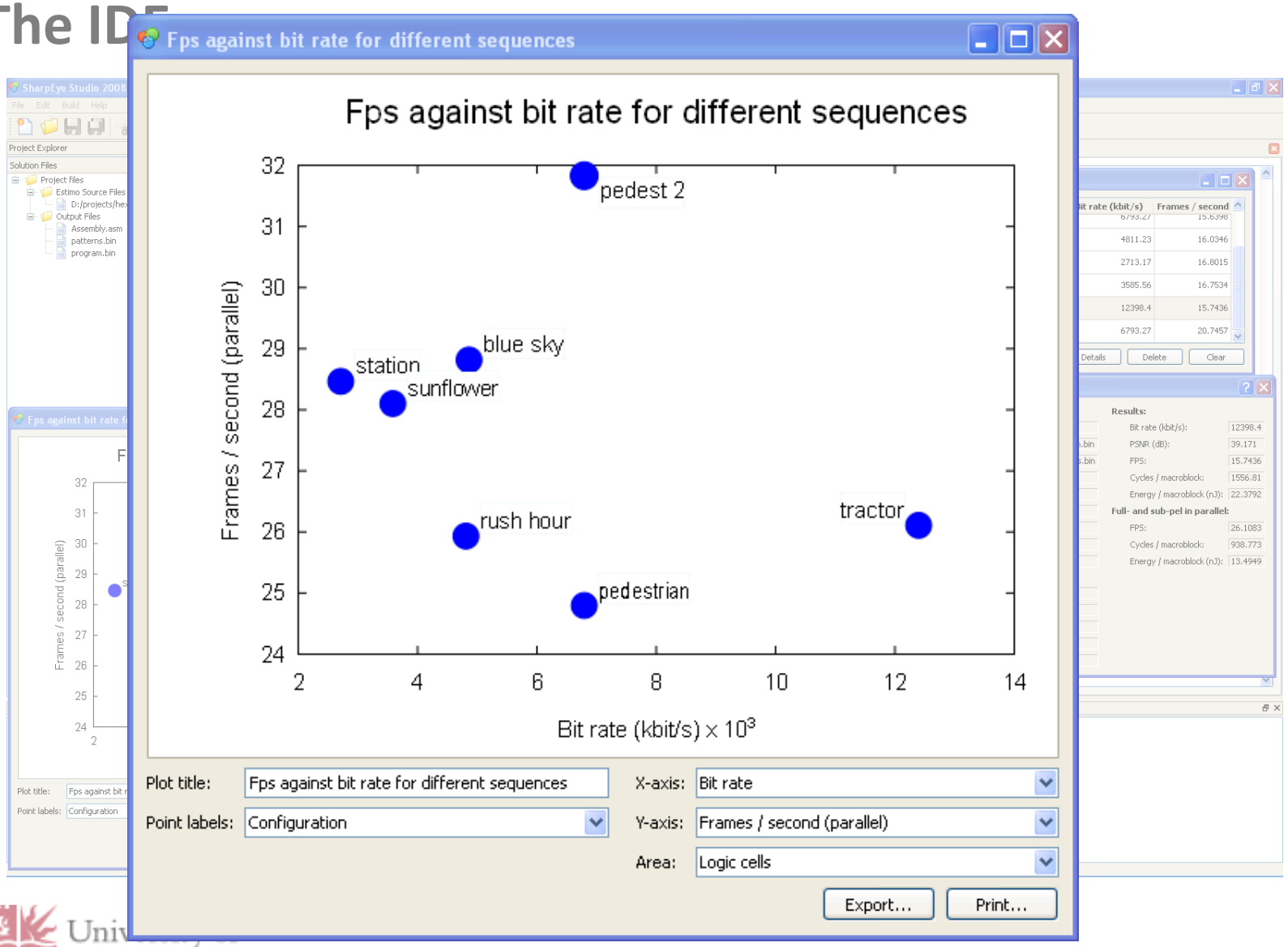
The IDE


SharpEye Studio 2008 - A complete framework for the Development of Motion Estimation Algorithms
File Edit Build Help
Project Explorer
Solution Files
Project files
Estimo Source Files
D:/projects/hex/hex.est
Output Files
Assembly.asm
patterns.bin
program.bin
hex.est
Pattern(quarter) {
check(0.25, 0)
check(0, 0.25)
check(-0.25, 0)
check(0, -0.25)
}
check(zero);
for(i = 1 to 8 step 1) {
check(hex);
#if(WINID == 0)
#break;
}
check(square);
for(i = 1 to 2 step 1) {
check(half);
#if(WINID == 0)
#break;
Cycle Accurate Model
Configuration
PSNR (dB)
Bit rate (kbit/s)
Frames / second
pedestrian 41.3 6793.27 15.6398
rush hour 41.874 4811.23 16.0346
station 40.705 2713.17 16.8015
sunflower 42.326 3585.56 16.7534
tractor 39.171 12398.4 15.7436
pedest 2 41.3 6793.27 20.7457
Rename... Export configuration... Details Delete Clear
Cycle Accurate Model
Motion-estimation search configuration
Program memory: D:/projects/hex/estimo.output/program.bin Browse...
Points memory: D:/projects/hex/estimo.output/patterns.bin Browse...
Full-pel execution units: 2
Sub-pel execution units: 1
Smallest partition: 16x16
Enable Motion Vector cost optimization
Enable Motion Vector candidates
Video data
Video file: D:/test_sequences/1080p/pedestrian_area.yuv Browse...
Resolution: 1920 x 1080
Maximum number of frames: 50
QP (0 is lossless): 26
Reference frames: 1
Results
Processor configuration:
Label: Cfg 24
Full-pel units: 2
Sub-pel units: 1
Smallest partition: 16x16
MV cost optimization: enabled
MV candidates: enabled
Logic cells: 11287
Video data:
Frames processed: 50
QP: 26
Reference frames: 1
Results:
Bit rate (kbit/s): 6793.27
PSNR (dB): 41.3
FPS: 20.7457
Cycles / macroblock: 1181.44
Energy / macroblock (nJ): 20.9706
Full- and sub-pel in parallel:
FPS: 31.8295
Cycles / macroblock: 770.033
Energy / macroblock (nJ): 13.6681
Run Stop New Plot Table
tractor
Processor configuration:
Label: tractor
Program memory: D:/projects/hex/estimo.output/program.bin
Point memory: D:/projects/hex/estimo.output/patterns.bin
Full-pel units: 1
Sub-pel units: 1
Smallest partition: 16x16
MV cost optimization: enabled
MV candidates: enabled
Logic cells: 9732
Video data:
Video file: D:/test_sequences/1080p/tractor.yuv
Resolution: 1920x1080
Frames processed: 50
QP (0 is lossless): 26
Reference frames: 1
Results:
Bit rate (kbit/s): 12398.4
PSNR (dB): 39.171
FPS: 15.7436
Cycles / macroblock: 1556.81
Energy / macroblock (nJ): 22.3792
Full- and sub-pel in parallel:
FPS: 26.1083
Cycles / macroblock: 938.773
Energy / macroblock (nJ): 13.4949
Fps against bit rate for different sequences
Fps against bit rate for different sequences
Frames / second (parallel)
Bit rate (kbit/s) x 10^3
pedest 2
blue sky
station
sunflower
rush hour
tractor
pedestrian
Plot title: Fps against bit rate for different sequences
Point labels: Configuration
X-axis: Bit rate
Y-axis: Frames / second (parallel)
Area: Logic cells
Export... Print...

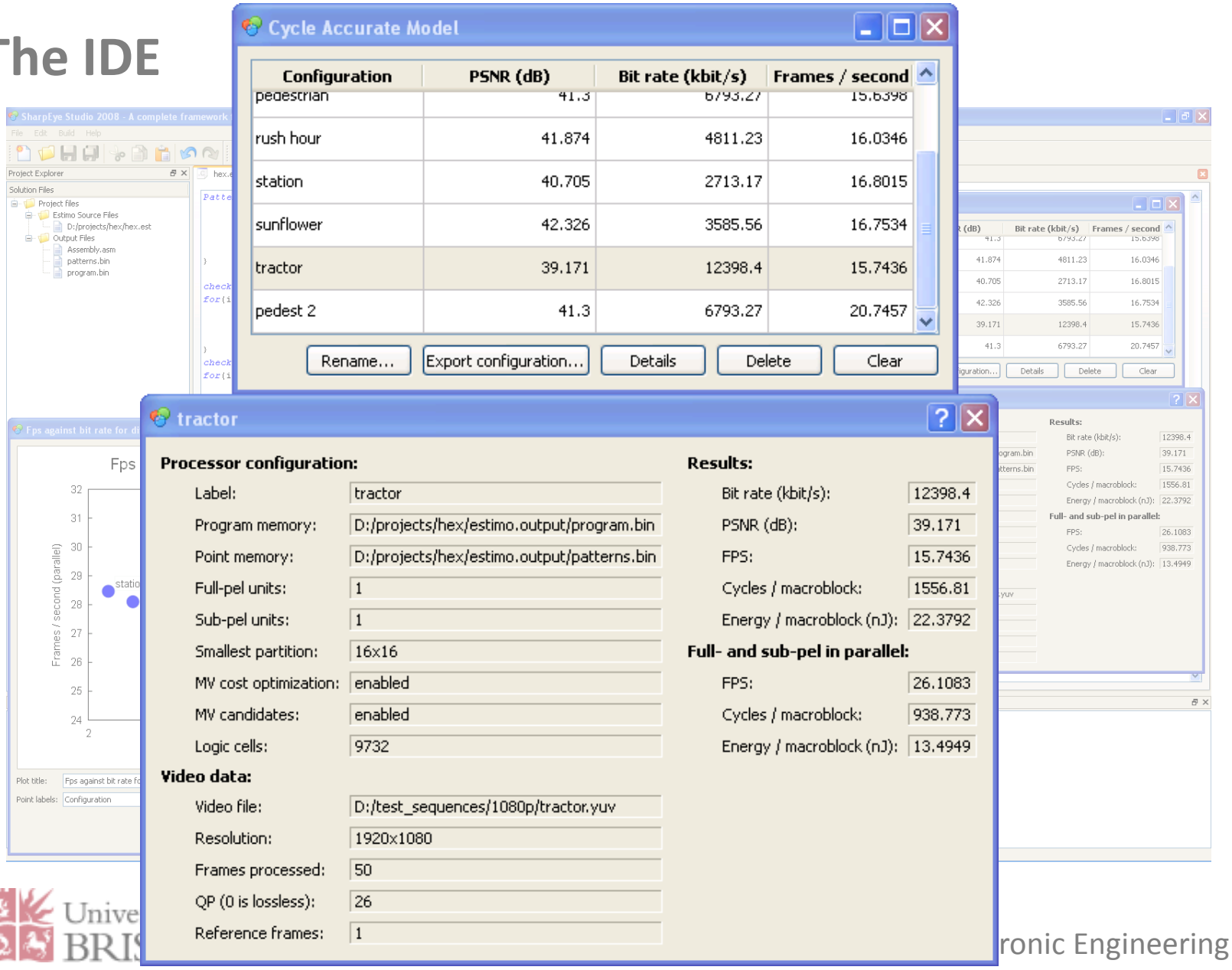
The IDE


Cycle Accurate Model
Motion-estimation search configuration
Program memory: D:/projects/hex/estimo.output/program.bin
Browse...
Points memory: D:/projects/hex/estimo.output/patterns.bin
Browse...
Full-pel execution units: 2
Sub-pel execution units: 1
Smallest partition: 16x16
Enable Motion Vector cost optimization
Enable Motion Vector candidates
Video data
Video file: D:/test_sequences/1080p/pedestrian_area.yuv
Browse...
Resolution: 1920 x 1080
Maximum number of frames: 50
QP (0 is lossless): 26
Reference frames: 1
Results
Processor configuration:
Label: Cfg 24
Full-pel units: 2
Sub-pel units: 1
Smallest partition: 16x16
MV cost optimization: enabled
MV candidates: enabled
Logic cells: 11287
Video data:
Frames processed: 50
QP: 26
Reference frames: 1
Results:
Bit rate (kbit/s): 6793.27
PSNR (dB): 41.3
FPS: 20.7457
Cycles / macroblock: 1181.44
Energy / macroblock (nJ): 20.9706
Full- and sub-pel in parallel:
FPS: 31.8295
Cycles / macroblock: 770.033
Energy / macroblock (nJ): 13.6681
Run
Stop
New Plot
Table
Fps against bit rate for different sequences
Frames / second (parallel)
station
blue sky
sunflower
rush hour

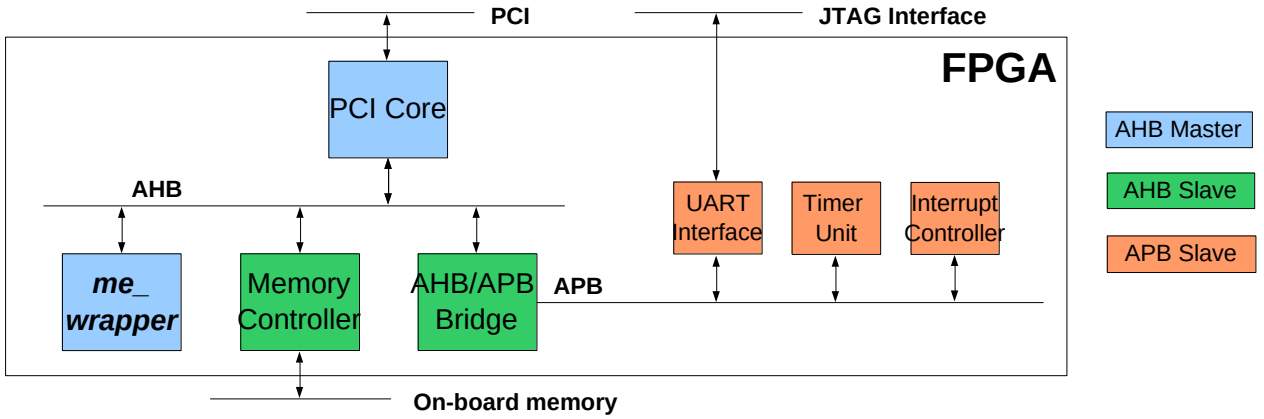
The IDE

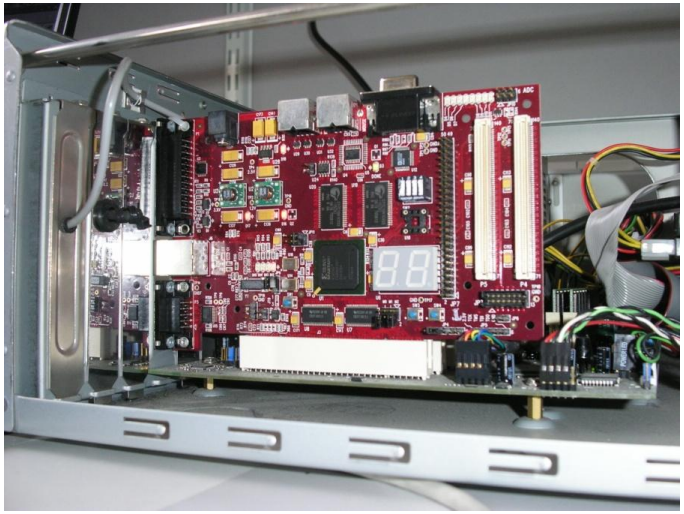

Fps against bit rate for different sequences
Fps against bit rate for different sequences
pedest 2
blue sky
station
sunflower
rush hour
tractor
pedestrian
Frames / second (parallel)
Bit rate (kbit/s) x 10^3
Plot title:
Fps against bit rate for different sequences
Point labels:
Configuration
X-axis:
Bit rate
Y-axis:
Frames / second (parallel)
Area:
Logic cells
Export...
Print...

The IDE


Cycle Accurate Model
Configuration
PSNR (dB)
Bit rate (kbit/s)
Frames / second
pedestrian 41.3 6793.27 15.6398
rush hour 41.874 4811.23 16.0346
station 40.705 2713.17 16.8015
sunflower 42.326 3585.56 16.7534
tractor 39.171 12398.4 15.7436
pedest 2 41.3 6793.27 20.7457
Rename...
Export configuration...
Details
Delete
Clear
tractor
Processor configuration:
Label: tractor
Program memory: D:/projects/hex/estimo.output/program.bin
Point memory: D:/projects/hex/estimo.output/patterns.bin
Full-pel units: 1
Sub-pel units: 1
Smallest partition: 16x16
MV cost optimization: enabled
MV candidates: enabled
Logic cells: 9732
Video data:
Video file: D:/test_sequences/1080p/tractor.yuv
Resolution: 1920x1080
Frames processed: 50
QP (0 is lossless): 26
Reference frames: 1
Results:
Bit rate (kbit/s): 12398.4
PSNR (dB): 39.171
FPS: 15.7436
Cycles / macroblock: 1556.81
Energy / macroblock (nJ): 22.3792
Full- and sub-pel in parallel:
FPS: 26.1083
Cycles / macroblock: 938.773
Energy / macroblock (nJ): 13.4949

Prototype implementation





Summary

- Programmable and configurable processor supports HD motion estimation
  (supports H.264, MPEG-4, MPEG-2, VC-1, AVS).
- Motion Estimation Processor: http://www.opencores.org/
- Estimo C compiler for easy development of proprietary block-matching algorithms.
- FPGA-based PCI demonstration board available.
- Cycle-accurate simulator for quick evaluation and design space exploration.
- SharpEye IDE: http://sharpeye.borelspace.com/